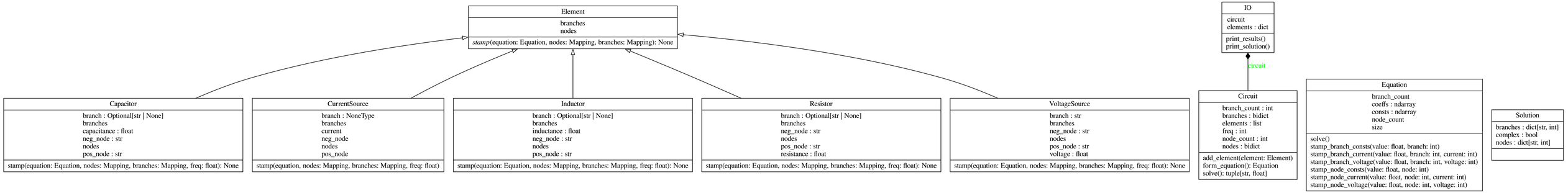
Element
branches
nodes
stamp(equation: Equation, nodes: Mapping, branches: Mapping): None
Capacitor
branch : Optional[str | None]
branches
capacitance : float
neg_node : str
nodes
pos_node : str
stamp(equation: Equation, nodes: Mapping, branches: Mapping, freq: float): None
CurrentSource
branch : NoneType
branches
current
neg_node
nodes
pos_node
stamp(equation, nodes: Mapping, branches: Mapping, freq: float)
Inductor
branch : Optional[str | None]
branches
inductance : float
neg_node : str
nodes
pos_node : str
stamp(equation: Equation, nodes: Mapping, branches: Mapping, freq: float): None
Resistor
branch : Optional[str | None]
branches
neg_node : str
nodes
pos_node : str
resistance : float
stamp(equation: Equation, nodes: Mapping, branches: Mapping, freq: float): None
VoltageSource
branch : str
branches
neg_node : str
nodes
pos_node : str
voltage : float
stamp(equation: Equation, nodes: Mapping, branches: Mapping, freq: float): None
IO
circuit
elements : dict
print_results()
print_solution()
circuit
Circuit
branch_count : int
branches : bidict
elements : list
freq : int
node_count : int
nodes : bidict
add_element(element: Element)
form_equation(): Equation
solve(): tuple[str, float]
Equation
branch_count
coeffs : ndarray
consts : ndarray
node_count
size
solve()
stamp_branch_consts(value: float, branch: int)
stamp_branch_current(value: float, branch: int, current: int)
stamp_branch_voltage(value: float, branch: int, voltage: int)
stamp_node_consts(value: float, node: int)
stamp_node_current(value: float, node: int, current: int)
stamp_node_voltage(value: float, node: int, voltage: int)
Solution
branches : dict[str, int]
complex : bool
nodes : dict[str, int]